



Minería de datos y Patrones

Version 2024-I

Classification - Intro

[Capítulo 4]

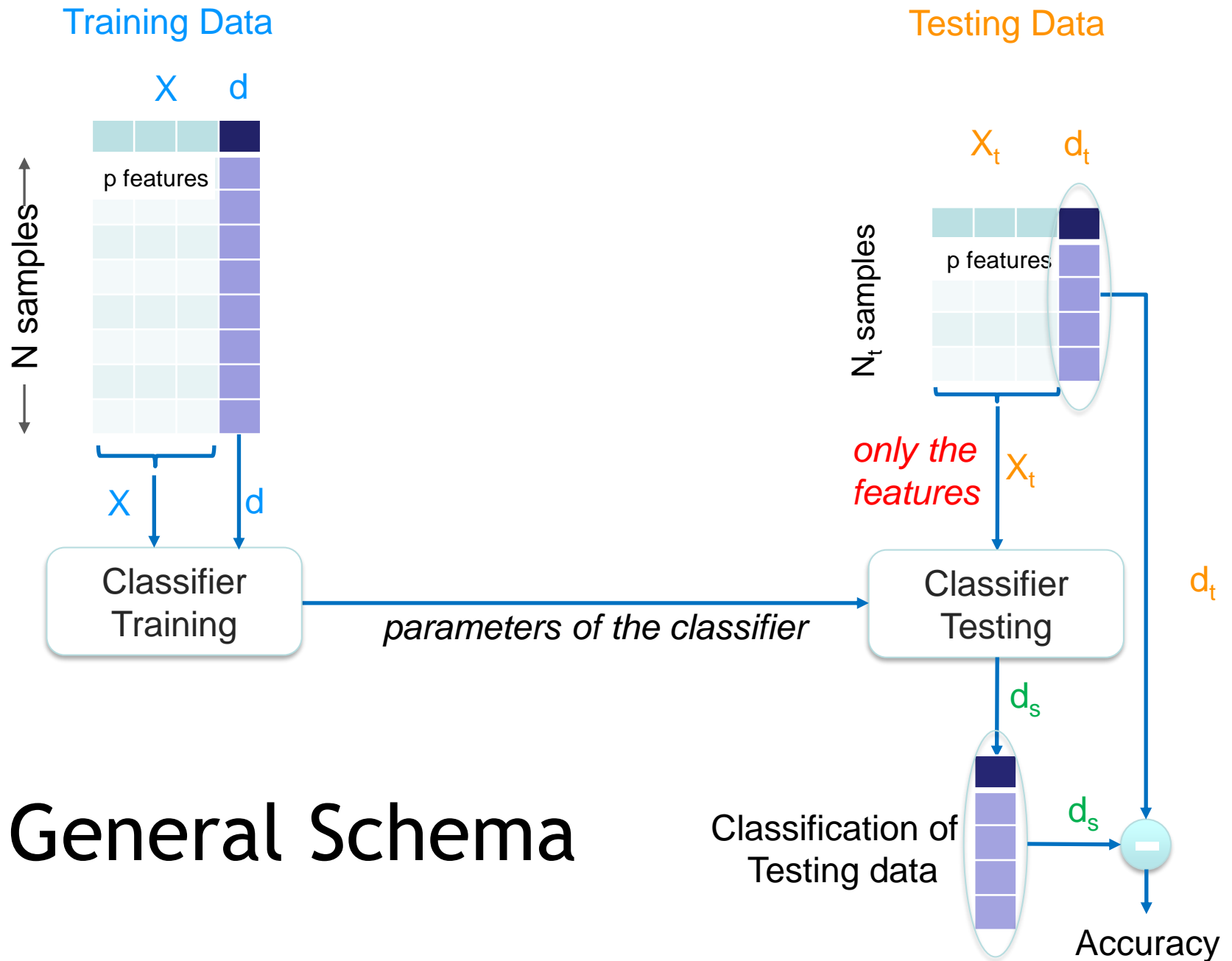
Dr. José Ramón Iglesias

DSP-ASIC BUILDER GROUP

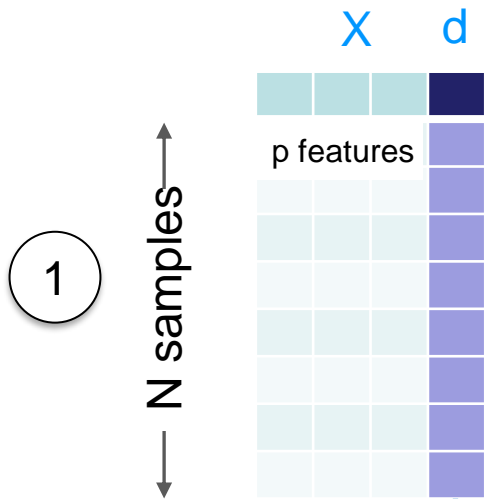
Director Semillero TRIAC

Ingeniería Electronica

Universidad Popular del Cesar



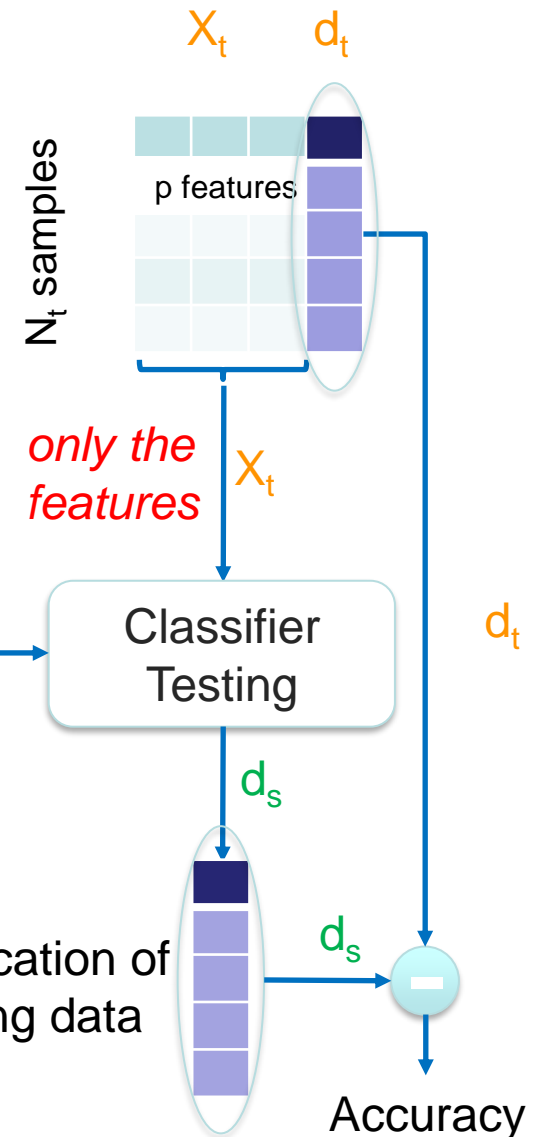
Training Data



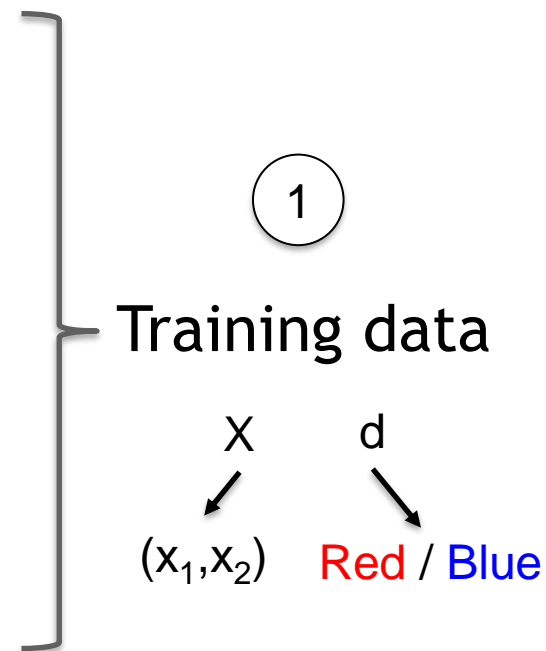
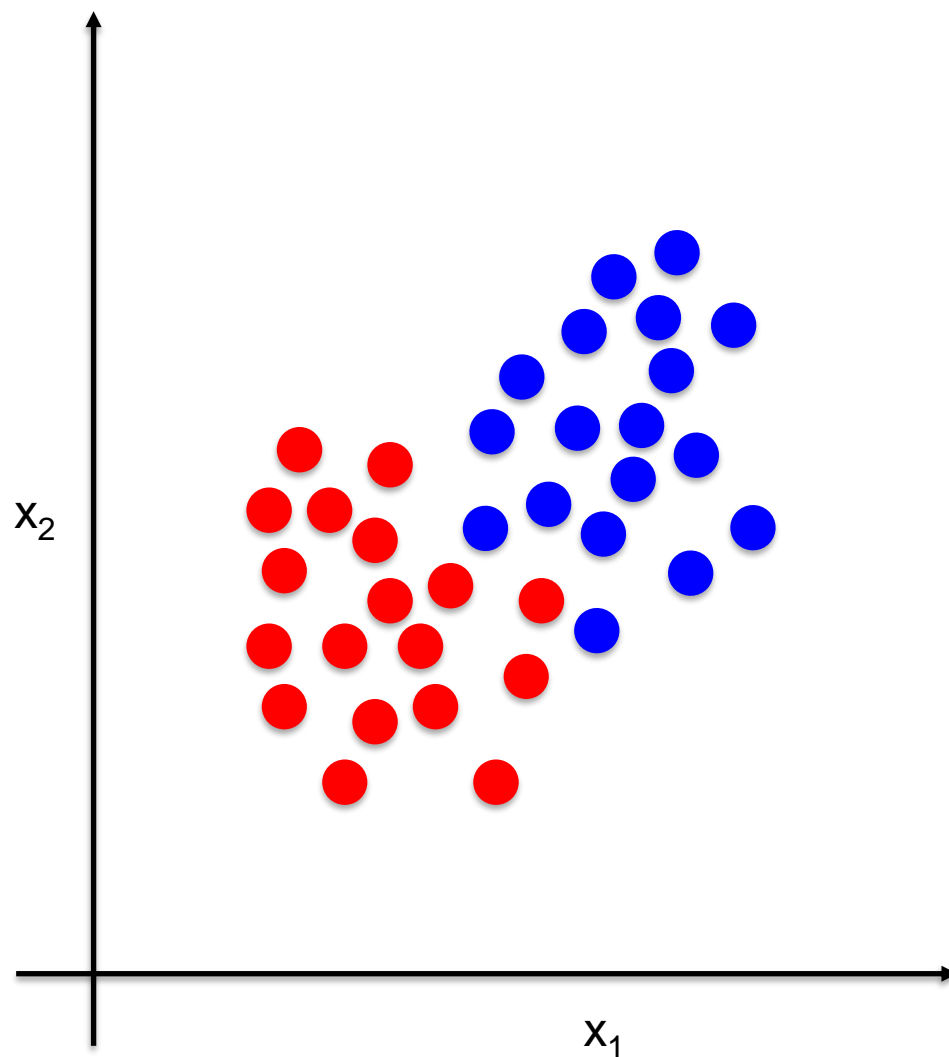
Classifier
Training

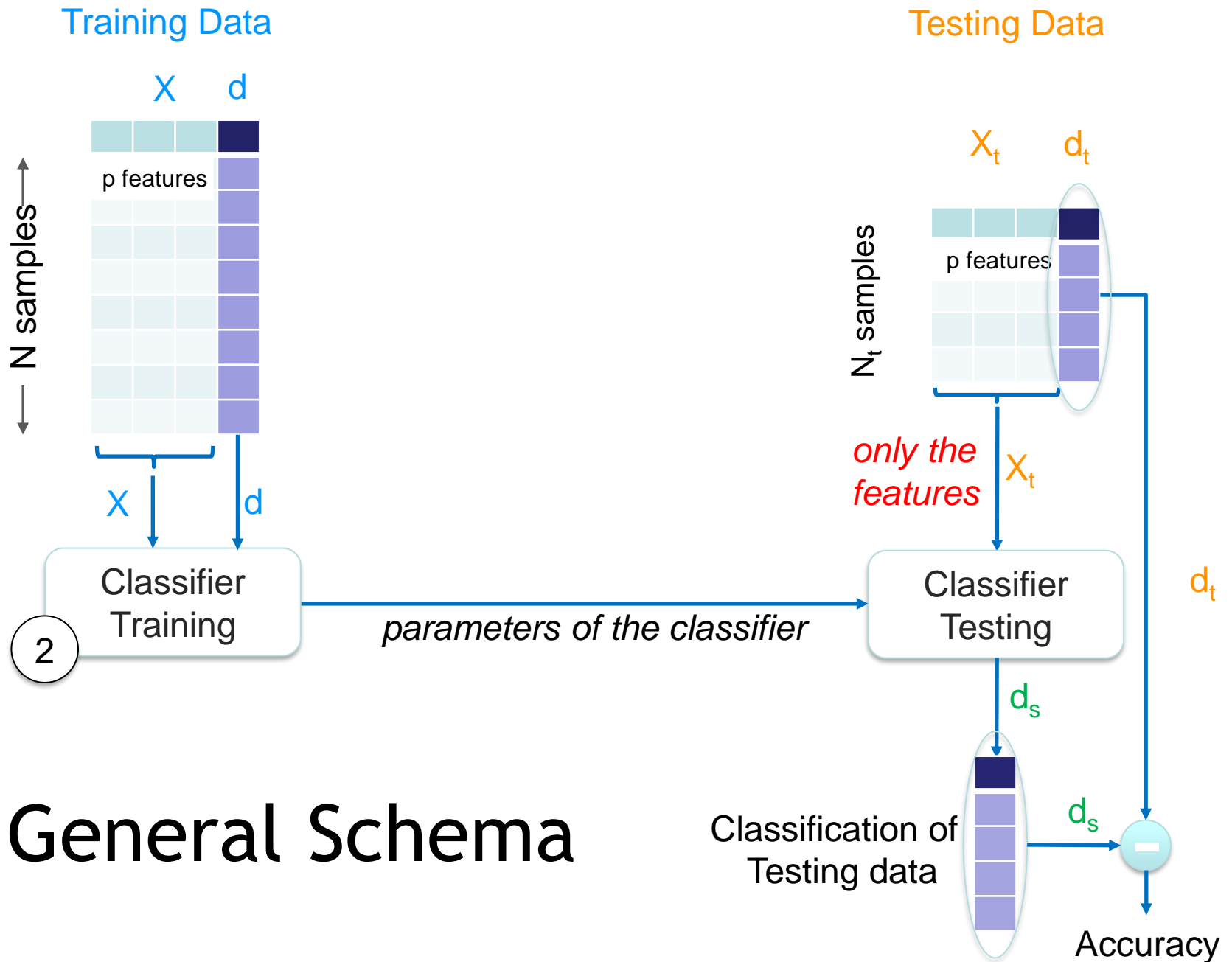
parameters of the classifier

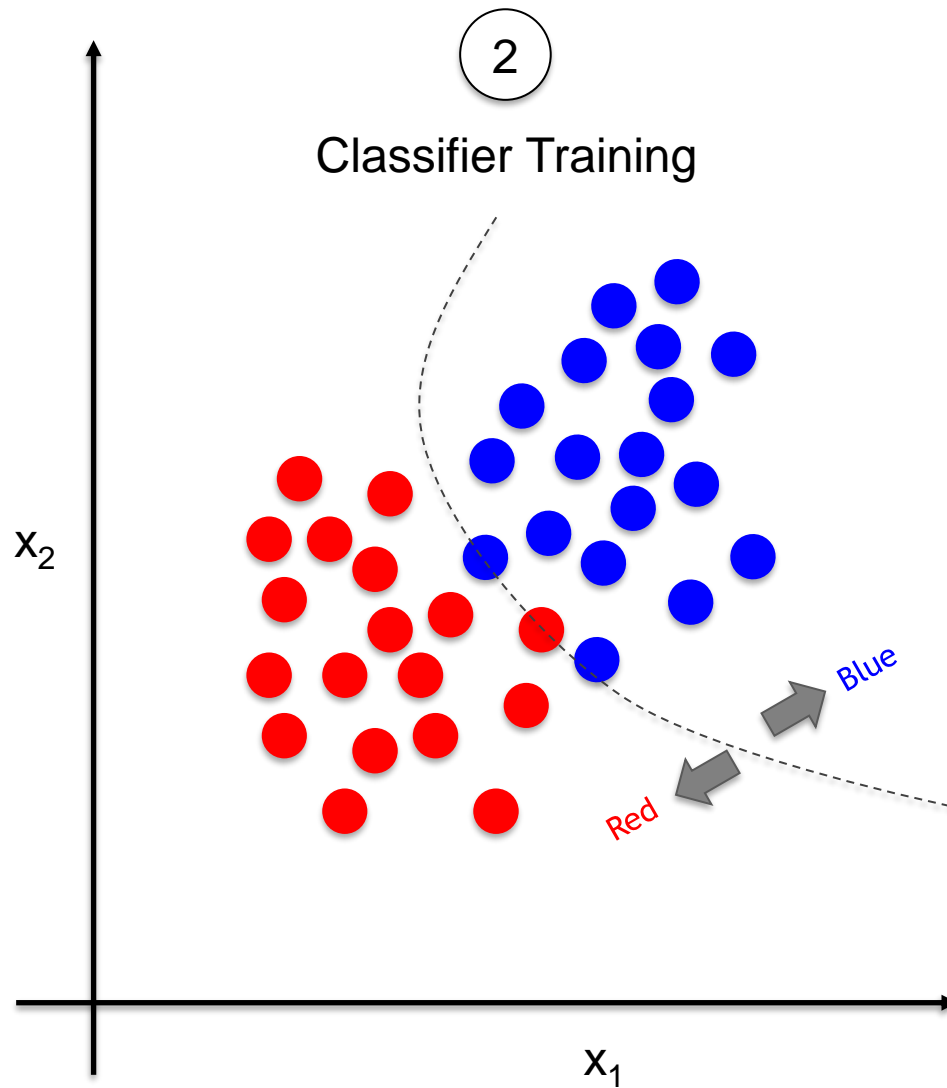
Testing Data

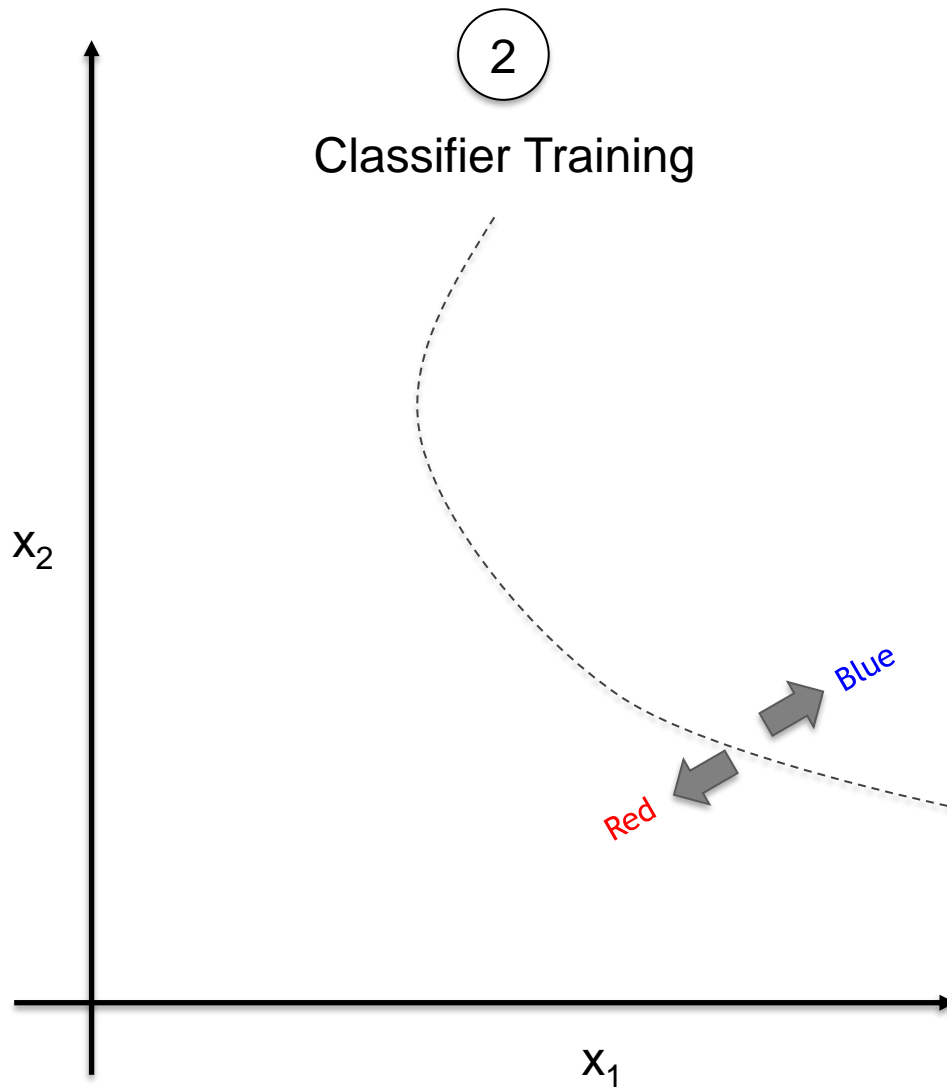


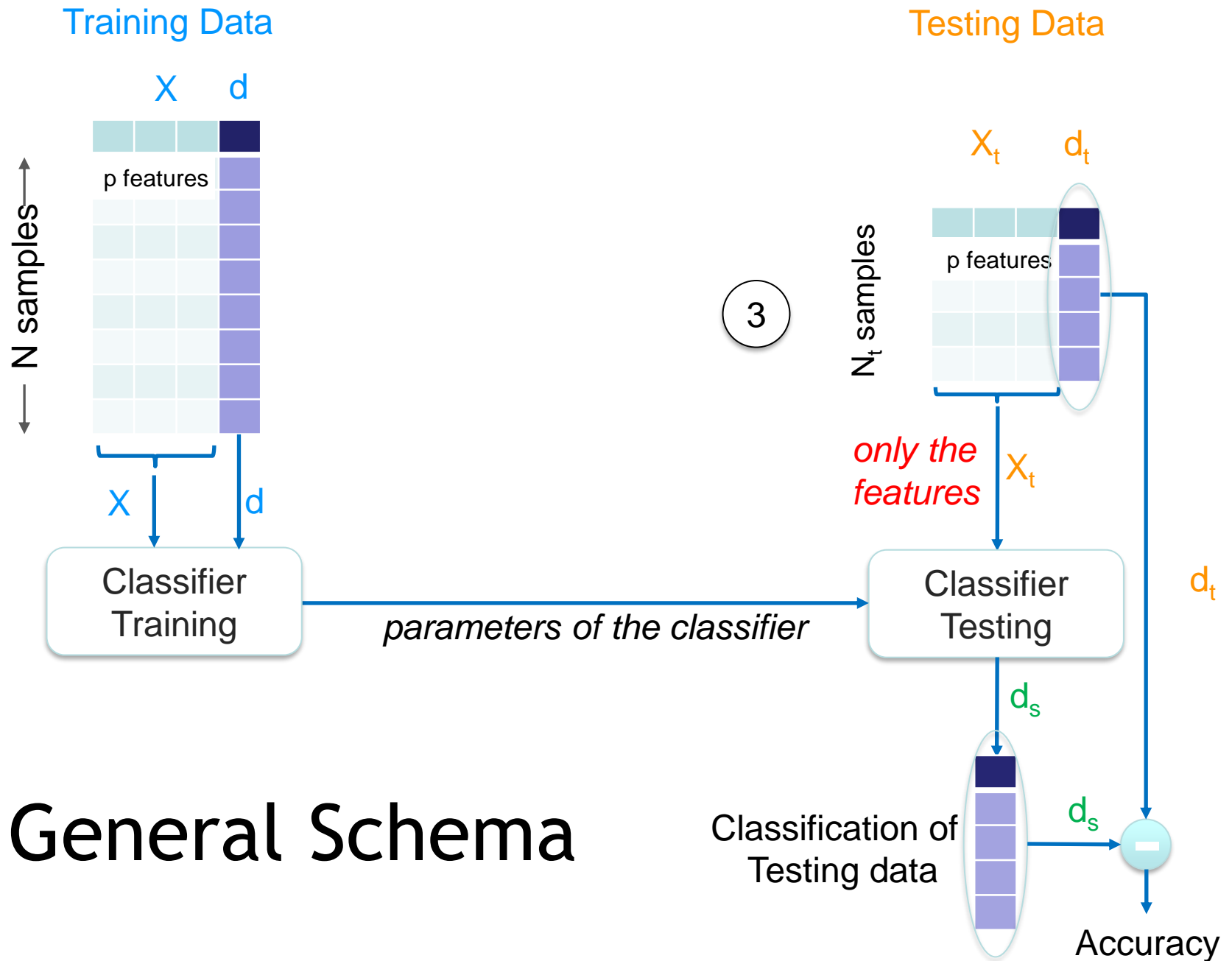
General Schema

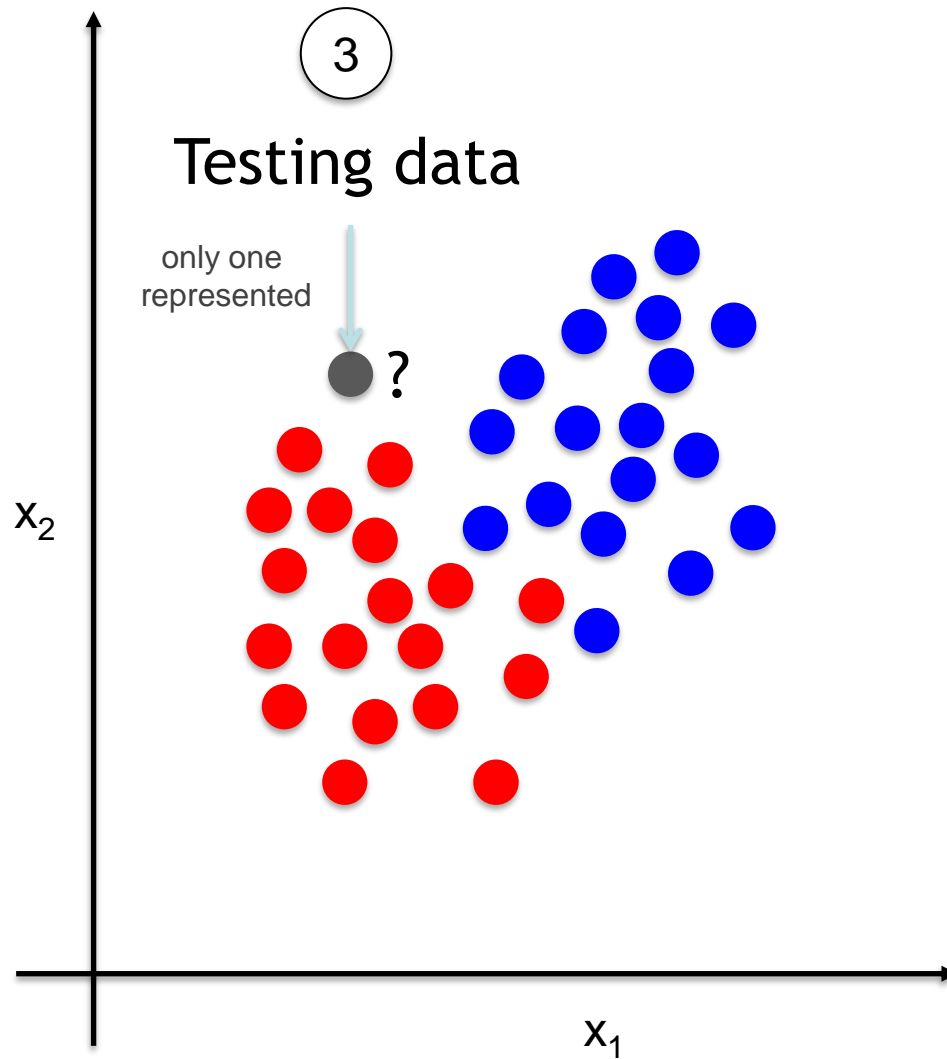


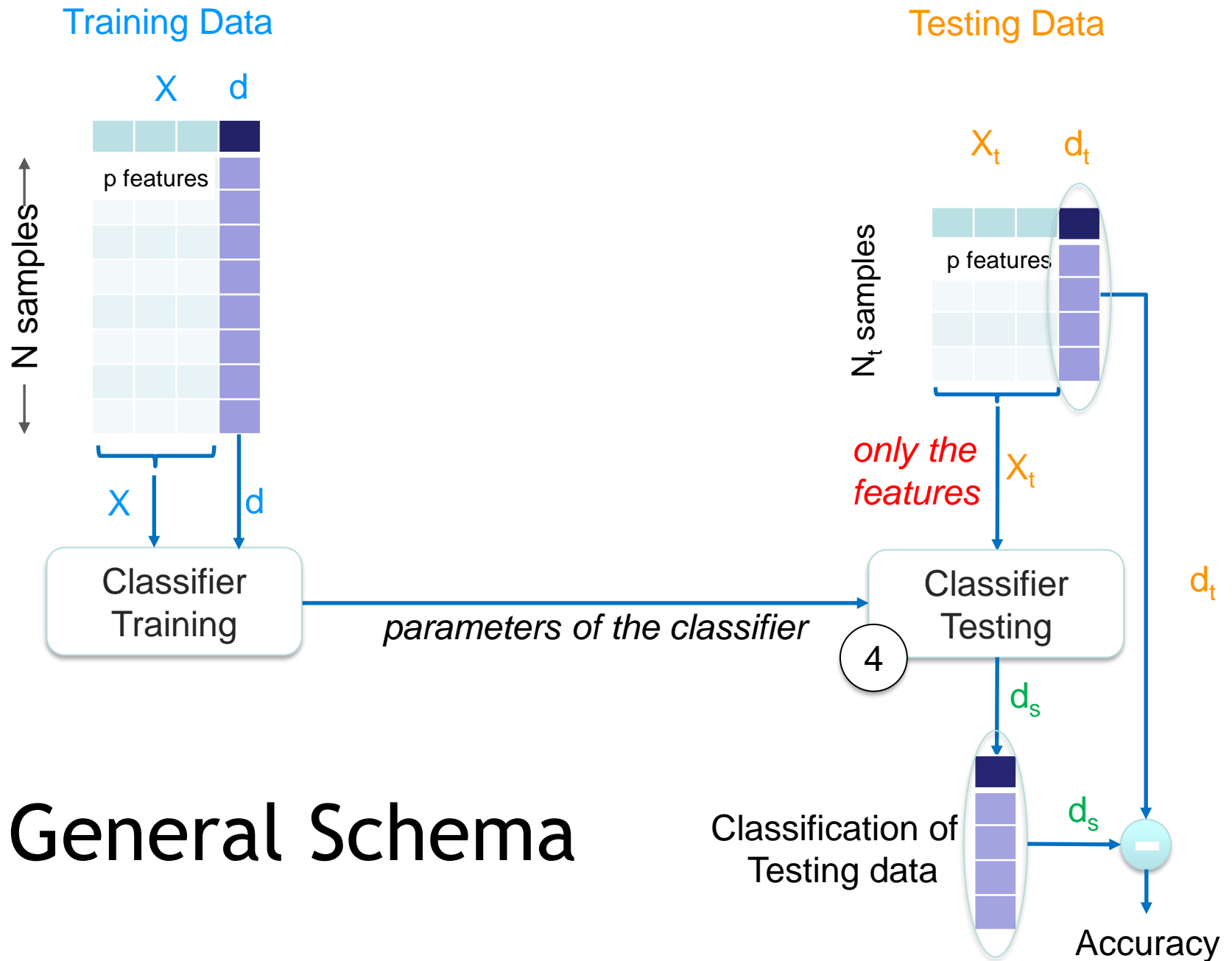


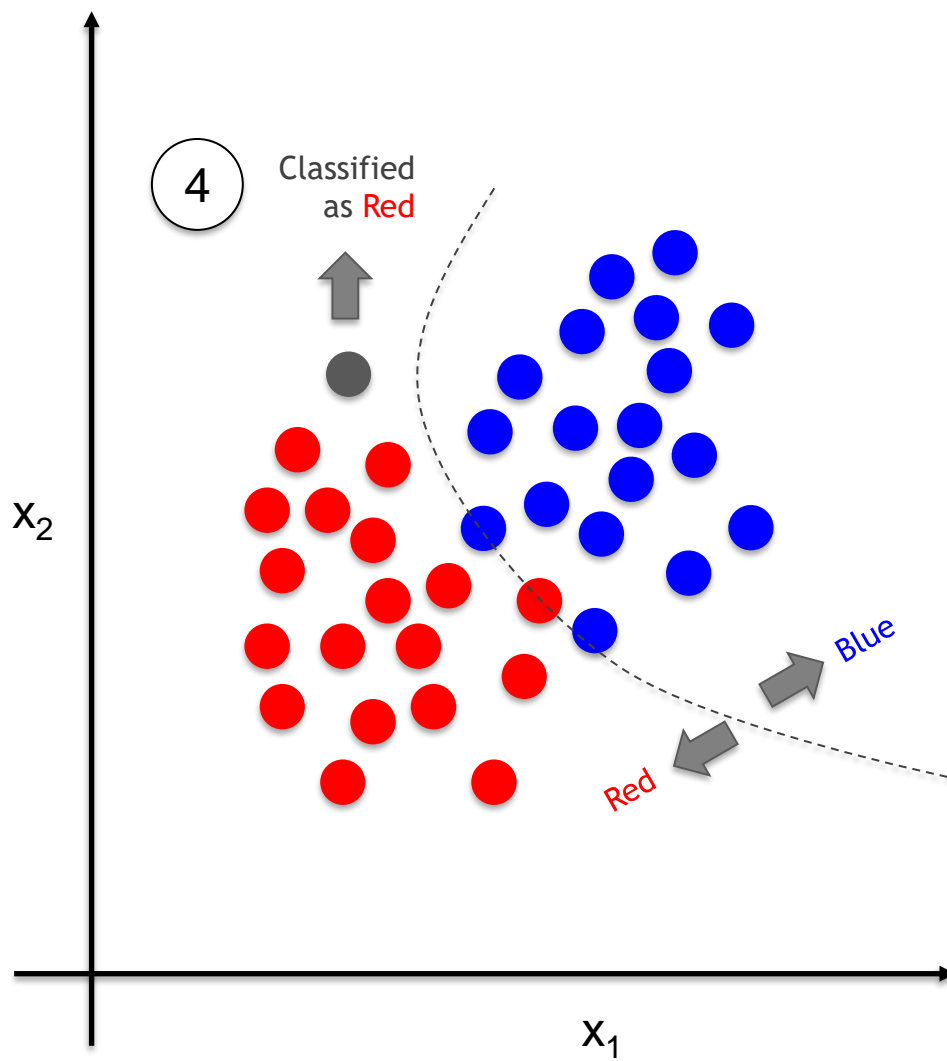


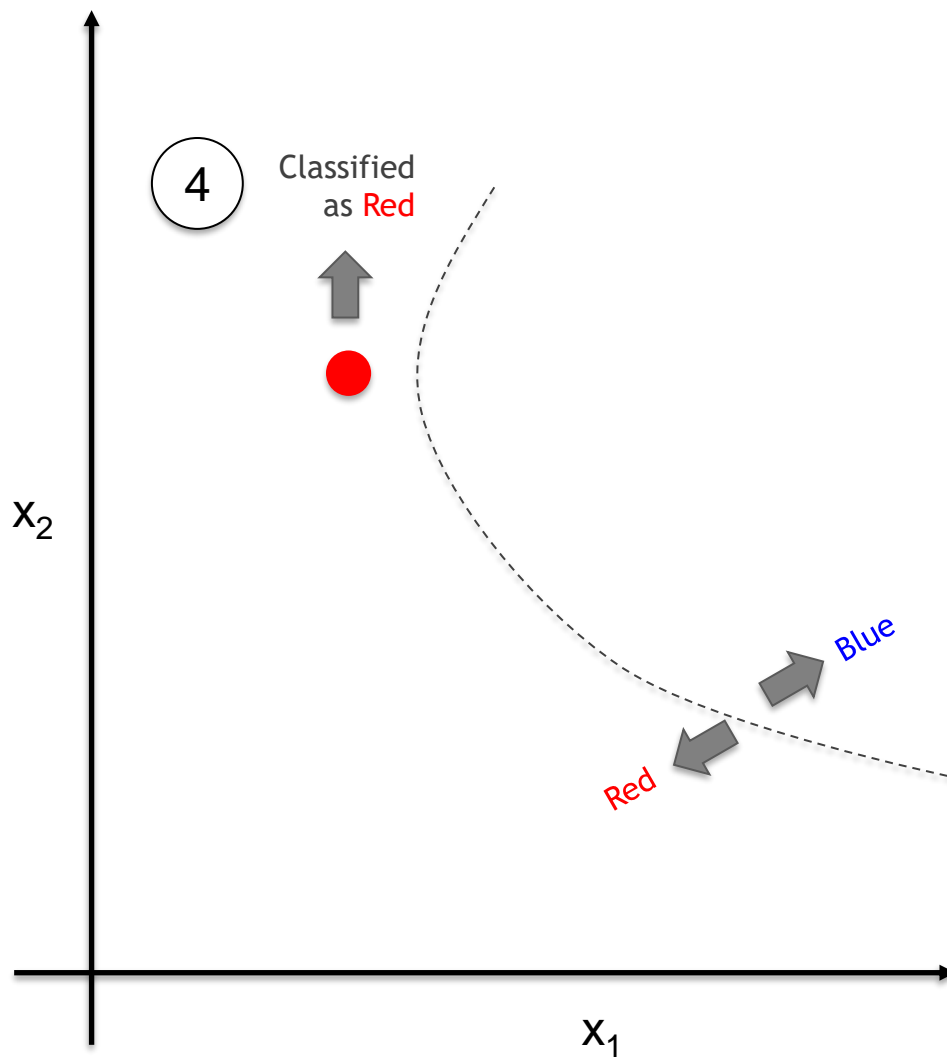


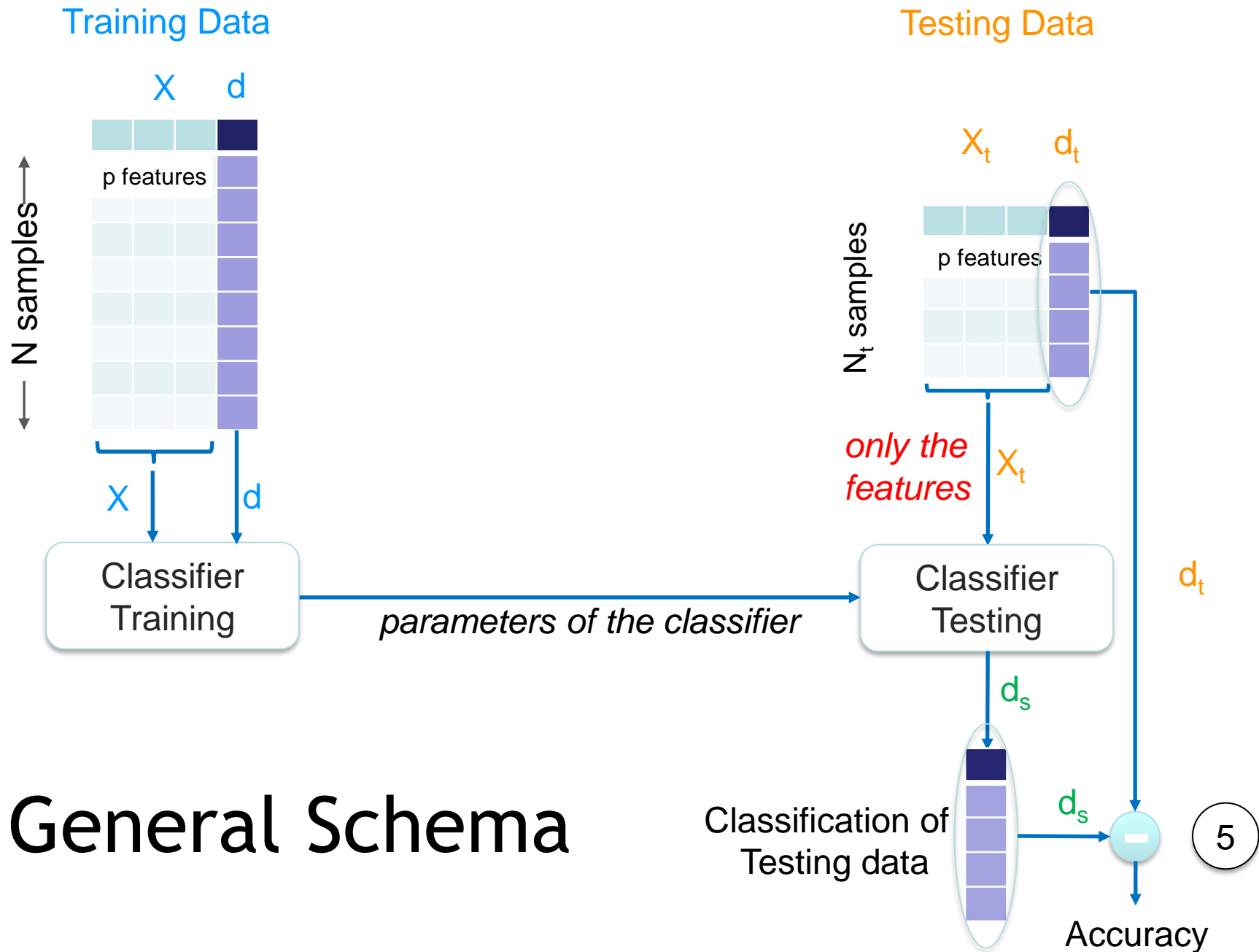


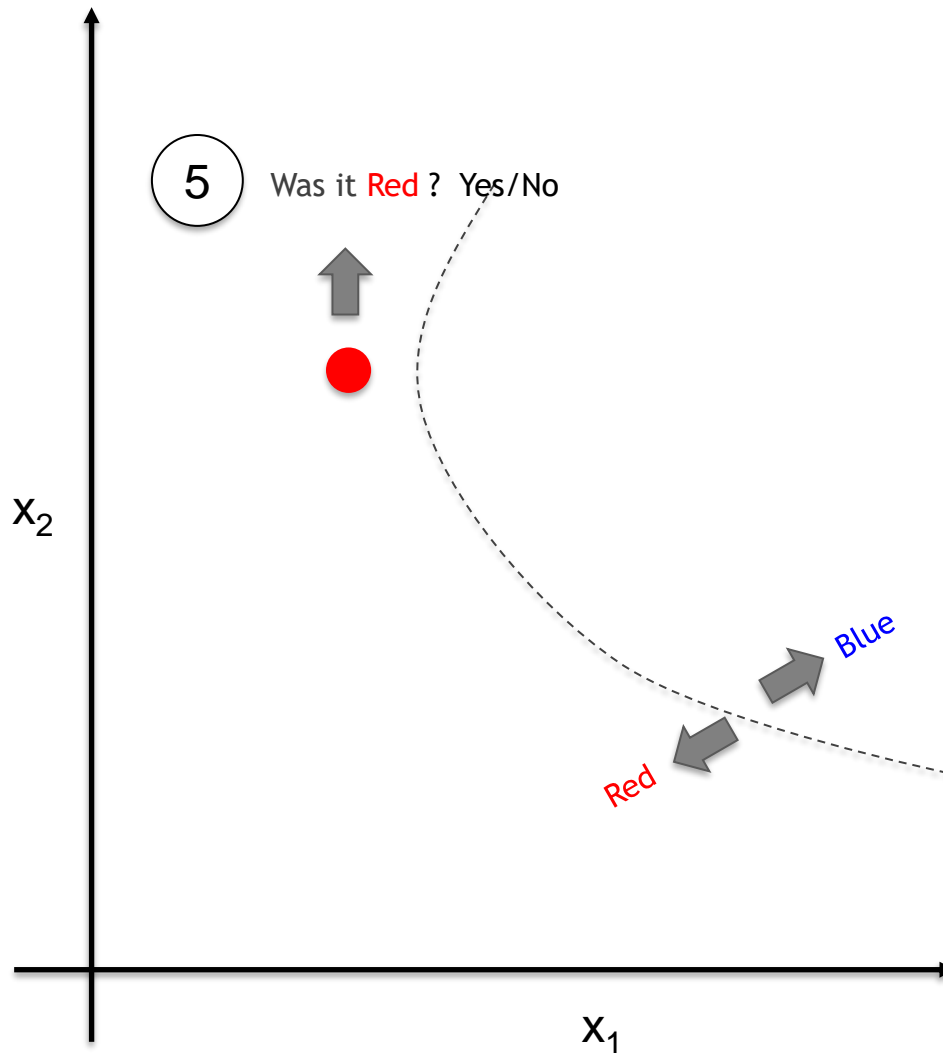


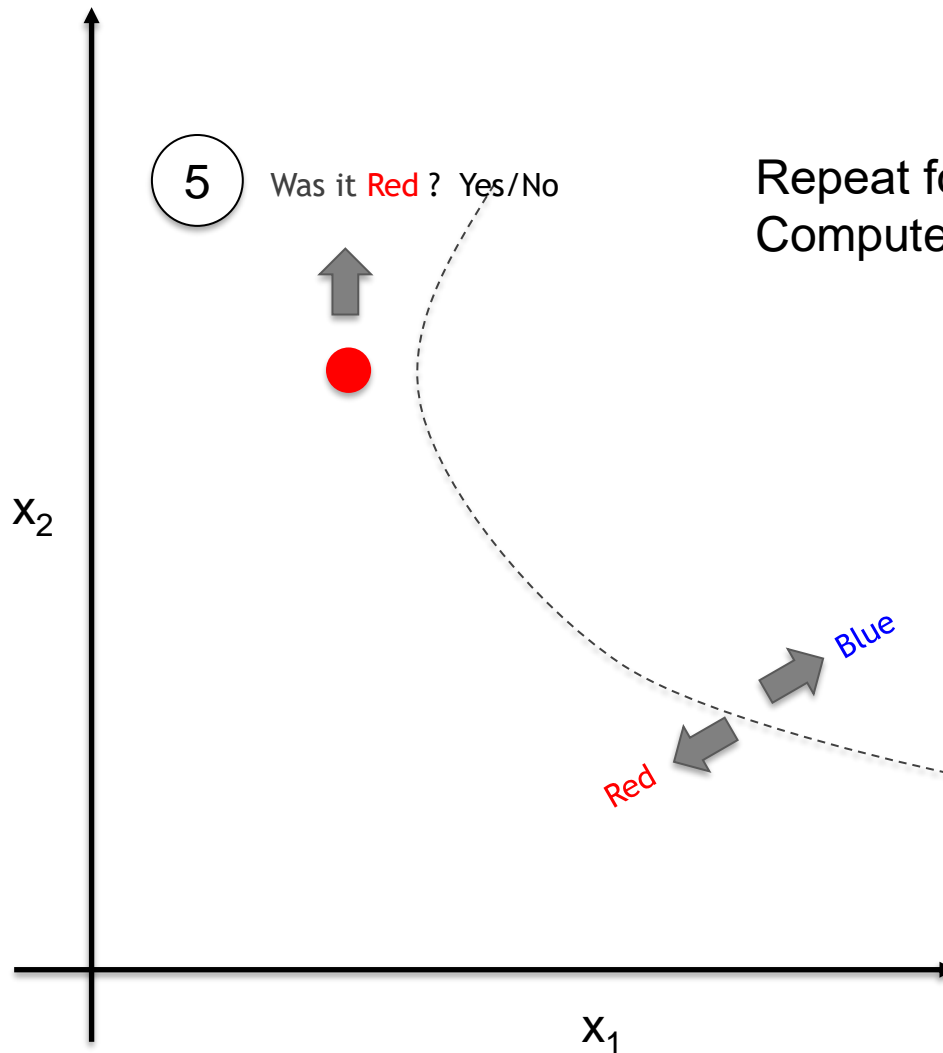










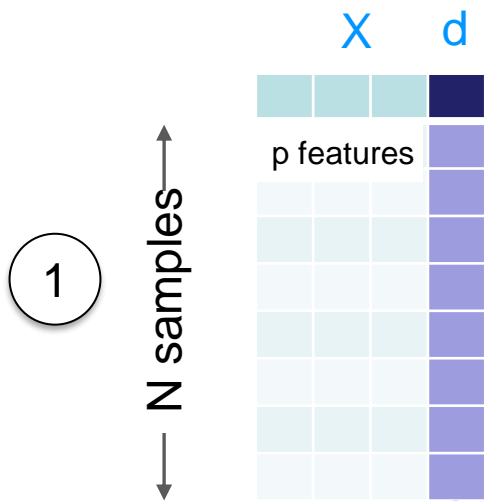


5

Was it Red? Yes/No

Repeat for all testing samples...
Compute Accuracy

Training Data

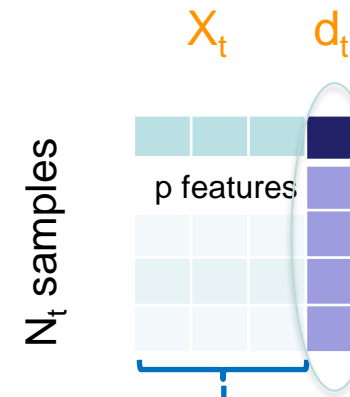


2

Classifier Training

parameters of the classifier

Testing Data



only the features

X_t

4

Classifier Testing

d_t

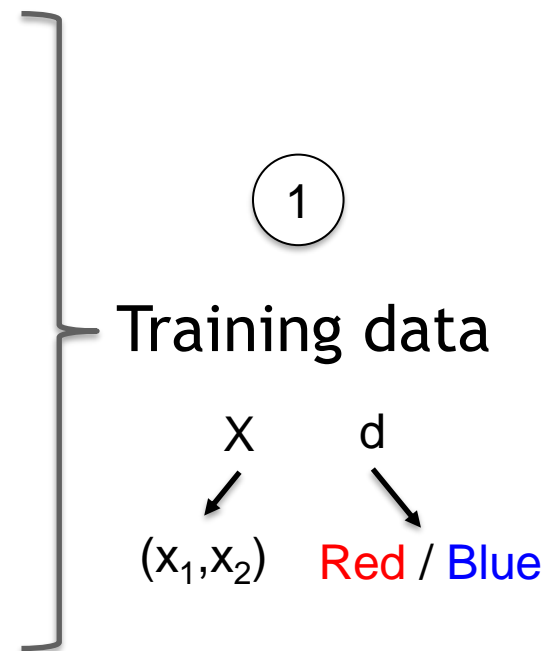
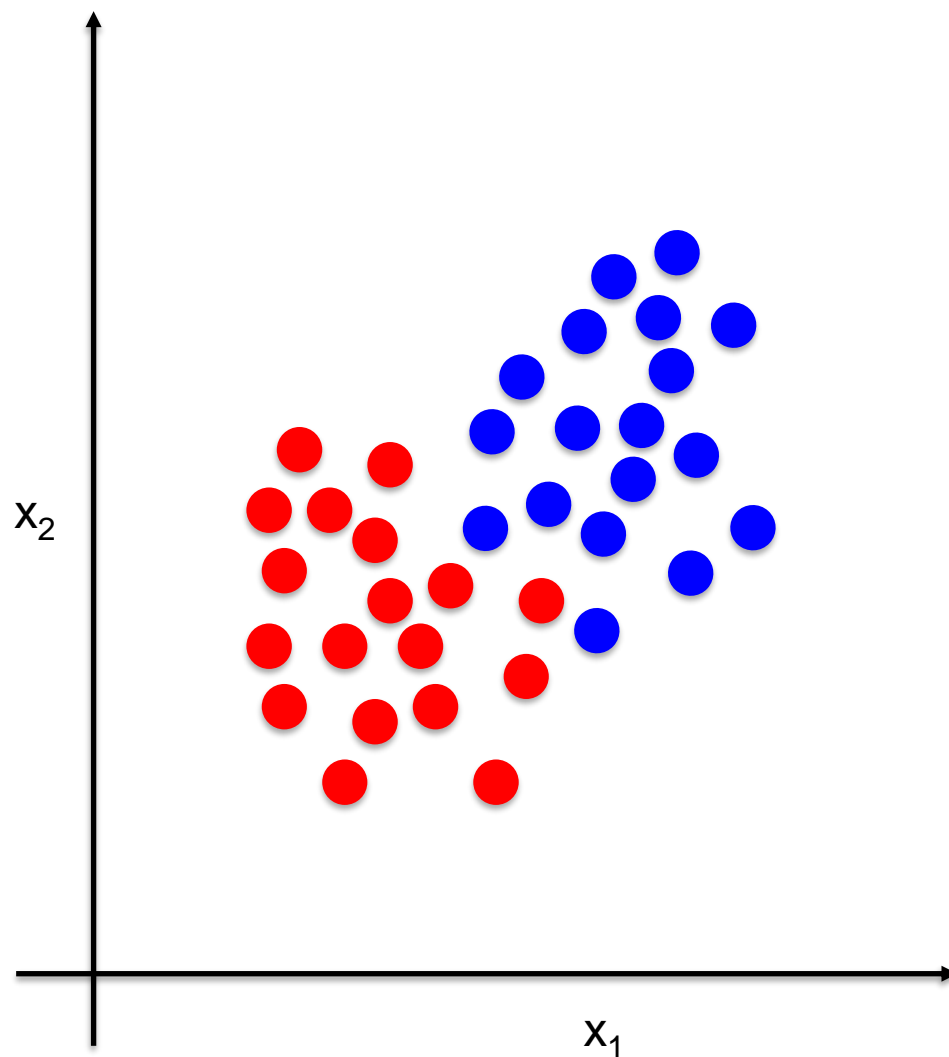
d_s

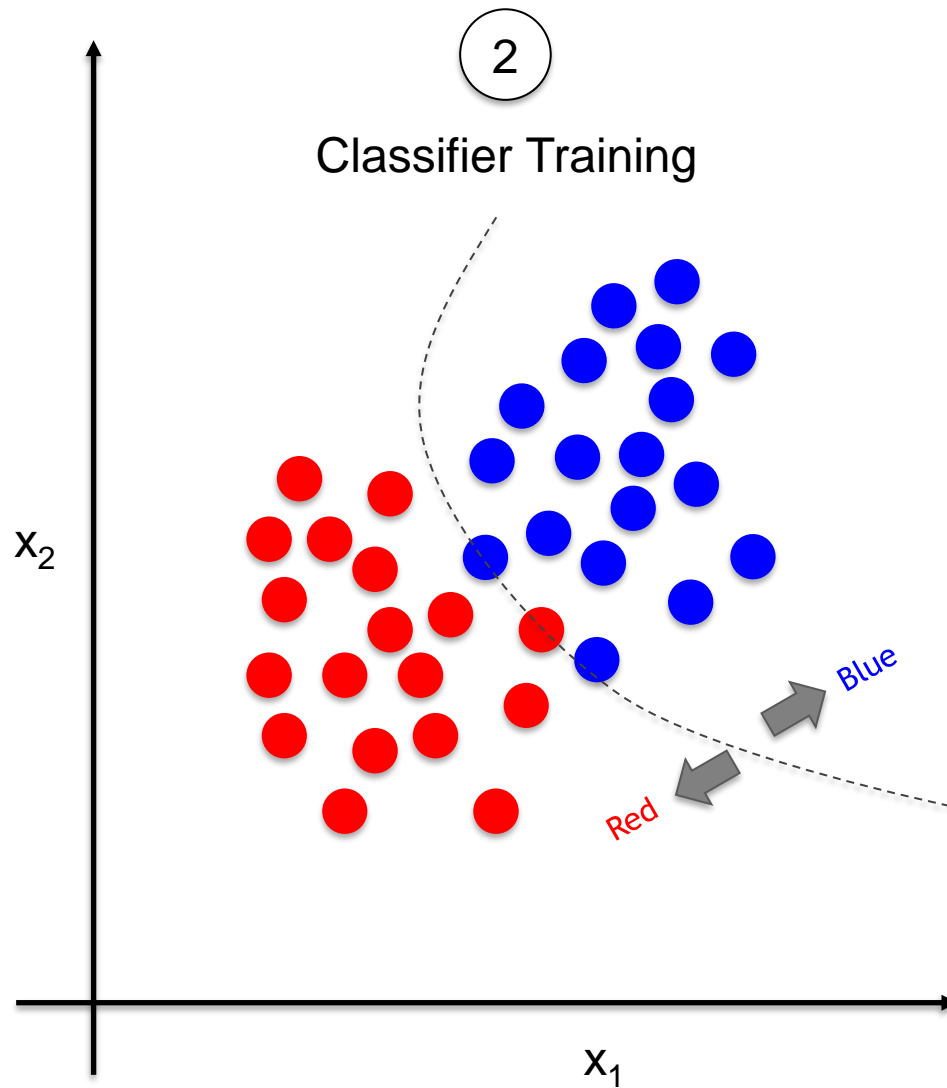
Classification of Testing data

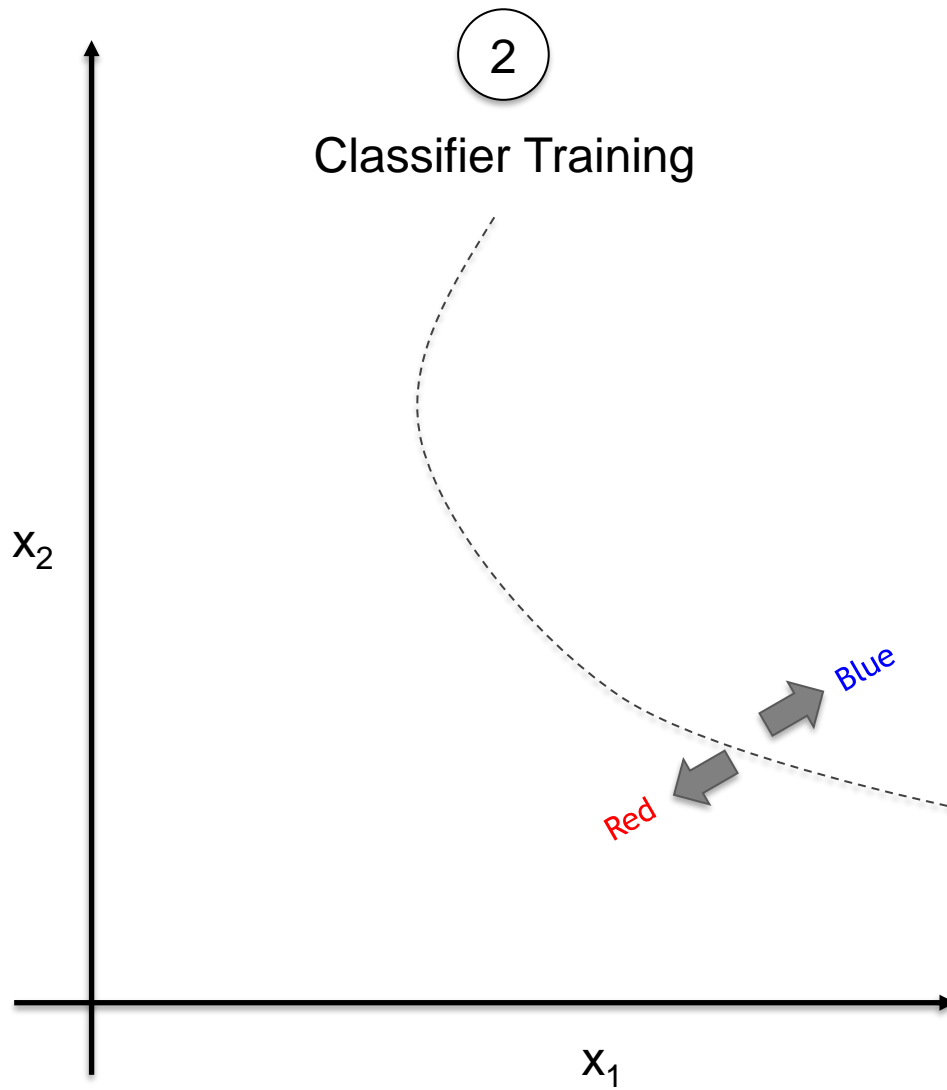
Accuracy

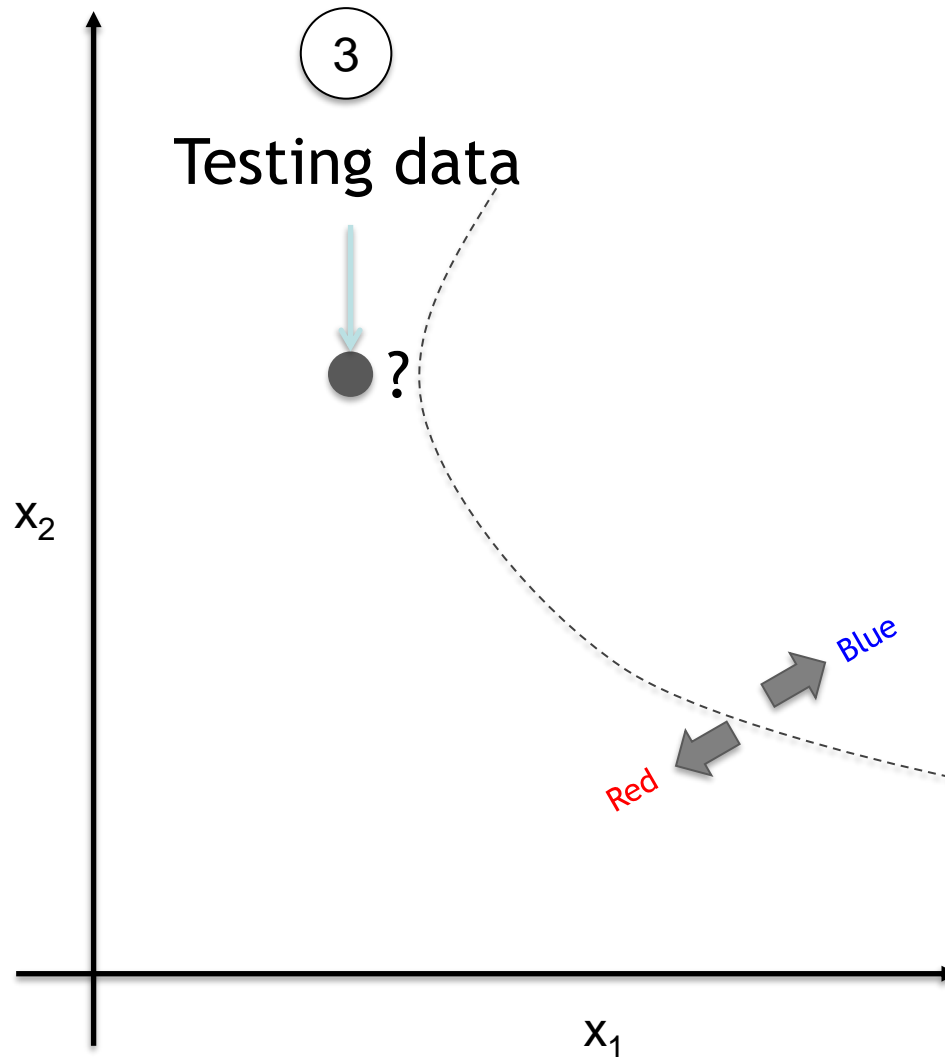
5

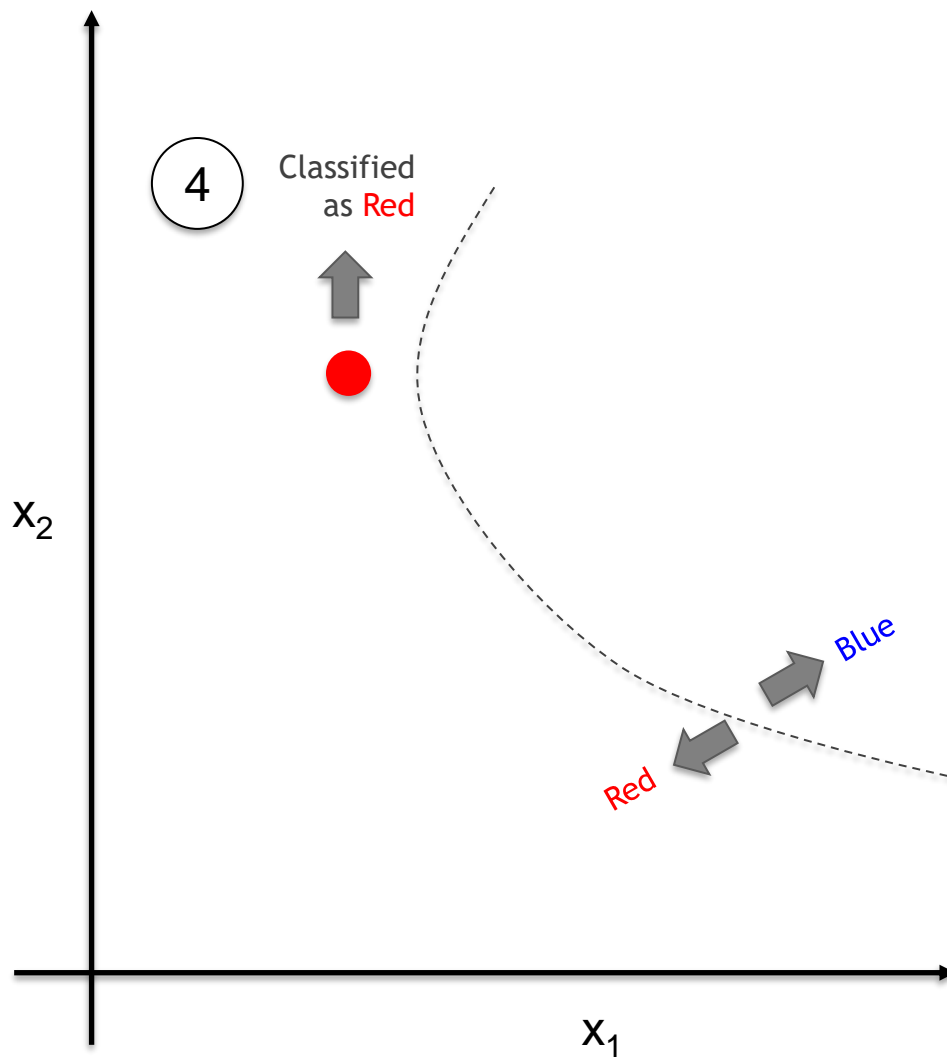
General Schema

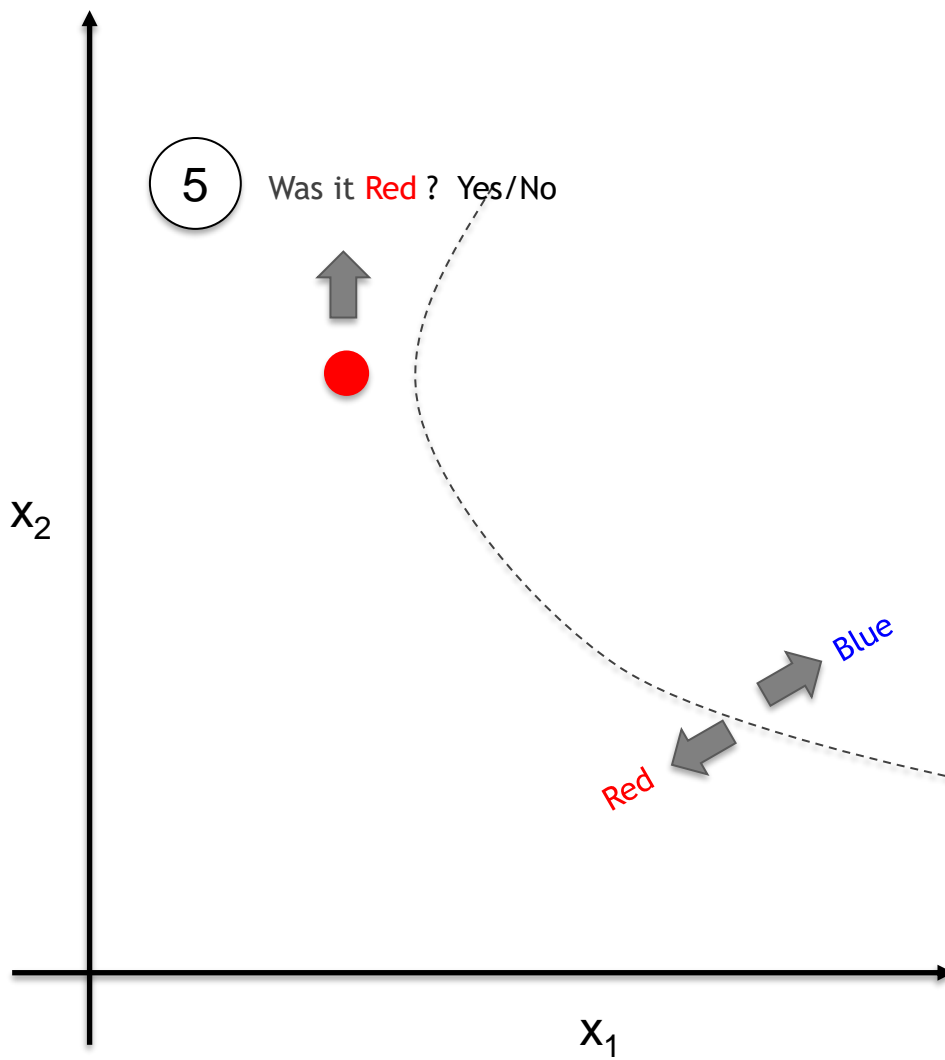


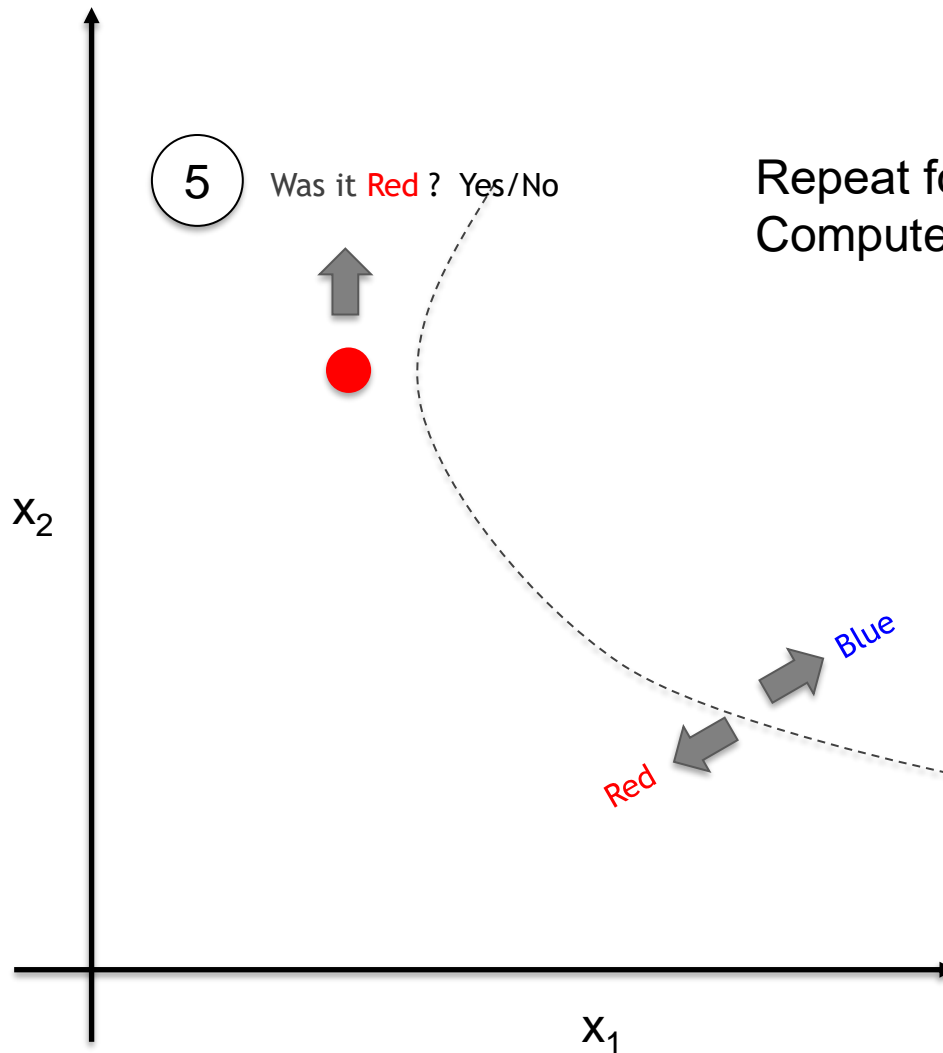










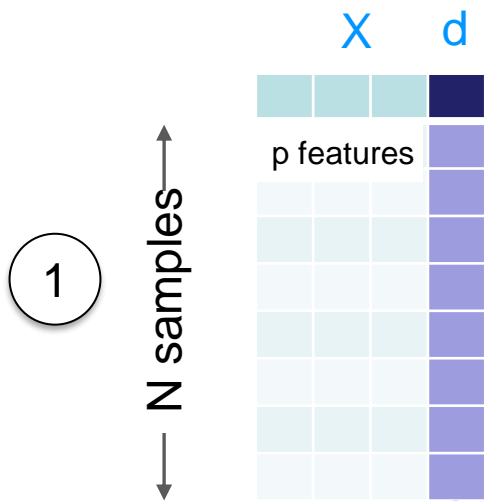


5

Was it Red? Yes/No

Repeat for all testing samples...
Compute Accuracy

Training Data

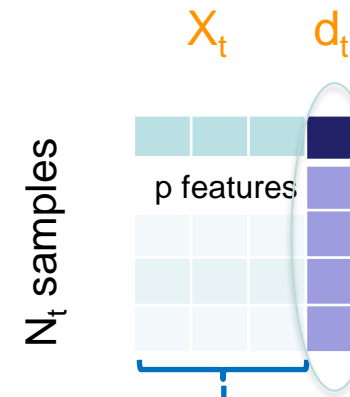


2

Classifier Training

parameters of the classifier

Testing Data



only the features

X_t

4

Classifier Testing

d_t

d_s

Classification of Testing data

Accuracy

5

General Schema

Libraries

STEPS:

0. (X, d) and (X_t, d_t) ← read training and testing data
1. *Definition* ← Classifier Definition
2. *TrainedParameters* ← Training($X, d, \text{Definition}$)
3. d_s ← Testing($X_t, \text{TrainedParameters}$)
4. Accuracy ← Accuracy(d_s, d_t)

$ypred$

Example for KNN-3

```
1. clf = KNeighborsClassifier(n_neighbors=3)
2. clf.fit(Xtrain, ytrain)
3. ypred = clf.predict(Xtest)
4. acc = accuracy_score(ypred, ytest)
```